TECHNICAL REPORT ARBRL-TR-02346

COMPUTER ALGORITHMS FOR THE DESIGN AND IMPLEMENTATION OF LINEAR PHASE FINITE IMPULSE RESPONSE DIGITAL FILTERS

James N. Walbert

July 1981

US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND
BALLISTIC RESEARCH LABORATORY
ABERDEEN PROVING GROUND, MARYLAND

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER TECHNICAL REPORT ARBRL-TR-02346	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) COMPUTER ALGORITHMS FOR THE DESIGN AND IMPLEMENTATION OF LINEAR PHASE FINITE IMPULSE RESPONSE DIGITAL FILTERS		5. TYPE OF REPORT & PERIOD COVERED TECHNICAL REPORT 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) James N. Walbert		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS U.S. Army Ballistic Research Laboratory ATTN: DRDAR-BLP Aberdeen Proving Ground, MD 21005		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 1L161102AH80
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Armament Research & Development Command U.S. Army Ballistic Research Laboratory ATTN: DRDAR-BL Aberdeen Proving Ground, MD 21005		12. REPORT DATE JULY 1981 13. NUMBER OF PAGES 61
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Software	Differentiation	Phase Delay
Digital Filter	Passband	
Linear Phase	Stopband	
Finite Impulse Response	Transition band	
Hilbert Transform	Convolution	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number) jmk

A FORTRAN program, published in the open literature, for the design of linear phase finite impulse response digital filters has been installed on the BRL CDC computer. Portions of this program have been extracted and combined to form a subroutine for filter design. Ancillary subroutines have been developed to assist in the formulation of filter design parameters. A subroutine for convolution of data with digital filters of finite odd length has also been written.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

		Page
	LIST OF ILLUSTRATIONS	5
I.	INTRODUCTION	7
II.	A DESCRIPTION OF THE DIGITAL FILTER DESIGN PROGRAM	7
III.	A DESCRIPTION OF THE DIGITAL FILTER DESIGN SUBROUTINE	8
IV.	CONSIDERATIONS IN THE USE OF THE DESIGN SOFTWARE	8
V.	IMPLEMENTATION OF DIGITAL FILTERS	11
VI.	CONCLUSIONS	15
VII.	SUMMARY	15
VIII.	ACKNOWLEDGEMENTS	16
	REFERENCES	17
	APPENDIX A. A LISTING OF PROGRAM DESIGN	19
	APPENDIX B. A LISTING OF SUBROUTINE FILTER	41
	APPENDIX C. SAMPLE OUTPUT FROM PROGRAM DESIGN	53
	DISTRIBUTION LIST	57

LIST OF ILLUSTRATIONS

Figure		Page
1.	Frequency response of a low pass filter.	9
2.	Frequency response of a low pass filter with narrow transition band.	10
3.	Graphical construction of new end points for $\left\{ x_i \right\}$	13

I. INTRODUCTION

In 1973, McClellan and Parks[1,2] published a listing of a computer program for the design of finite-duration impulse-response digital filters. This program was unique in that the authors had developed a unified theory for the design of the four types of filters: bandpass, bandstop, Hilbert-transform, and differentiation. The resulting software is one of the most flexible digital filter design programs available.

In the analysis of ballistic data which has been converted from an analog voltage record to a digital time series, it is generally desirable to be able to isolate various signal components for individual study. Such components are usually identifiable by frequency content, and as a consequence, are ideally suited for separation or removal by digital filtering techniques. This report describes the adaptation of the filter design program to the CDC computer at BRL, the modification of a portion of this program into a subroutine, the development of subroutines to specify filter design parameters, and a convolution subroutine for filters of odd length. A complete description of design considerations for digital filters and definitions of related terms is beyond the scope of this report. Any of the cited references will provide the necessary information. This report does provide sufficient design information to allow the reader to implement digital filters; a subsequent BRL Technical Report will cover in greater detail specific application techniques.

II. A DESCRIPTION OF THE DIGITAL FILTER DESIGN PROGRAM

Only minor changes were made to the program as it appeared in reference 2. The program statement added was

```
PROGRAM DESIGN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7)
```

In the original program, when the value of the variable JPUNCH was input as 1, values of the filter coefficients were output to punched cards. In the program, as it exists on the CDC computer, TAPE7 may be specified in the jobstream to be any suitable device or file. The constants PI and PI2 (π and 2π, respectively) were extended to the full double precision word length for the CDC. The free-field input form of the original program was replaced with formatted input. Finally, a test for end-of-file on input was added to allow for multiple designs per computer run. A listing of program DESIGN is in Appendix A.

---

[1] J.H. McClellan, T.W. Parks, "A Unified Approach to the Design of Optimum FIR Linear-Phase Digital Filters," IEEE Trans. Circuit Theory, CT-20(6), 697-701 (1973).

[2] J.H. McClellan, T.W. Parks, L.R. Rabiner, "A Computer Program for Designing Optimum FIR Linear Phase Digital Filters, "IEEE Trans Audio Electroacoustics, AU-21(6), 506-526 (1973).

III. A DESCRIPTION OF THE DIGITAL FILTER DESIGN SUBROUTINE

For most applications to analysis of ballistic data, optimum digital filter design specifications are the result of a systematic trial-and-error investigation. Frequently, the design specifications change from one data event to the next because certain aspects of the experiment were non-repeatable. In view of these factors, it seemed appropriate to formulate a filter design subroutine for use in interactive analysis computer programs, thereby permitting tailoring of the filter design on a round-by-round basis.

Subroutine FILTER, a listing of which appears in Appendix B, is extracted from program DESIGN. It will design bandpass filters of up to 10 bands, but will not design Hilbert Transformers or differentiators. The grid density (LGRID) has been fixed at 16, but the subroutine otherwise retains the full flexibility of program DESIGN. All variable names used in program DESIGN are also retained.

The subroutine statement is

SUBROUTINE FILTER(NFILT,NBANDS,EDGE,FX,WTX,IPRINT,H), where NFILT is the filter length; NBANDS is the number of pass/stop bands; EDGE is an array containing the band edges, expressed as fractions of the sampling frequency; FX is an array containing the desired filter shape, (1. in the pass bands and 0. in the stop bands); WTX is an array containing the desired relative weighting in each band; IPRINT is a control variable for printing the coefficients (0-print coefficients, 1-don't print coefficients); and H is the array containing the filter coefficients on output. The variables NFILT, NBANDS, AND IPRINT are integers; the arrays EDGE,FX,WTX, and H are real, dimensioned 2*NBANDS, NBANDS, AND (NFILT+1)/2, respectively. If NFILT is even, then the H array is dimensioned NFILT/2.

IV. CONSIDERATIONS IN THE USE OF THE DESIGN SOFTWARE

For the purposes of this discussion, assume that the data sequence $\{x_i\}_{i=1}^{n}$ consists of points equally spaced in time; in particular, Δt will denote the time between two consecutive samples. The sampling frequency, f_s, is therefore $1/\Delta t$, and the bandwidth of the data is $.5f_s$. The bandwidth of the data represents the highest unaliased frequency present in the data, provided due care has been given to the sampling process.

The essence of the design algorithm is to approximate the desired filter response function on the frequency-amplitude plane from $-.5f_s$ to $+.5f_s$ on the frequency axis. The coefficients are designed in a normalized form on the interval [-.5, .5]. Moreover, the frequency response has either odd or even symmetry about the origin on the frequency axis, so that the design problem is completely determined by specifying the desired response on the normalized frequency interval [0., .5].

In Figure 1, below, is shown the frequency response of a typical low pass filter. This is a two band filter: it has a pass band and a stop band.

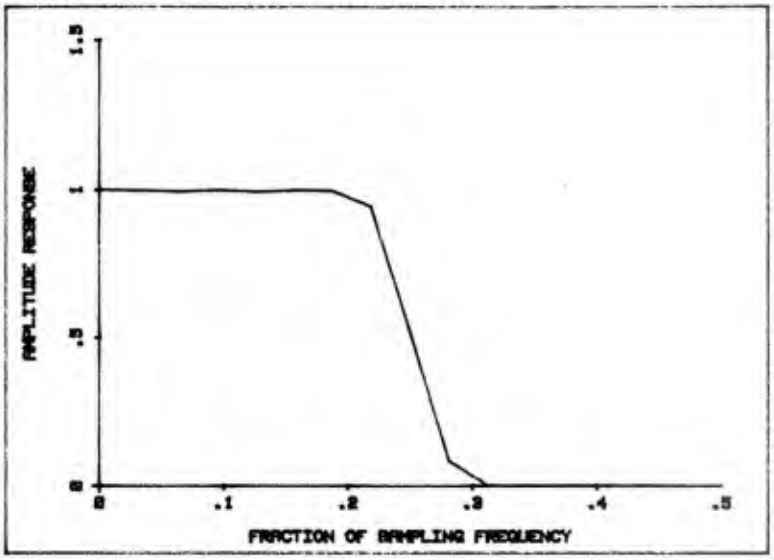

Figure 1. Frequency response of a low pass filter

The pass band is from 0. to .2, or to 40% of the bandwidth. The frequency $.2f_s$ is termed the cutoff frequency of the filter. It is a "pass" band since frequencies in this band are "passed" unaltered (i.e. are multiplied by 1). The stop band is from .3 to .5; frequencies in this band are "stopped" (i.e. multiplied by 0).

The frequency band from .2 to .3 is termed the transition band. Selection of the width of this transition band is somewhat critical in the design of a digital filter, for the following reason: as the transition band narrows, the slope of the frequency response (i.e. the filter roll-off) increases. As this slope increases, the design algorithm compensates by increasing the deviation from the desired response in the pass and stop bands. This deviation is called the "ripple", and results in increases and decreases of amplitude in the signal at those particular frequencies. An example of a filter designed with too narrow a transition band is shown in Figure 2.

In any application software, it is advisable to have the capability of viewing the frequency response of the filter prior to its application, in order to be certain of its characteristics. The design program, as a part of its printed output, lists the normalized frequencies at which the maximum and minimum amplitudes of the ripple occur. Also listed are the deviations from the desired design, which provide a measure of the amplitude error to be expected as a result of applying the filter to the data. (See Appendix C).

Referring to the example of Figure 1, the input variables to design this filter were assigned the following values:

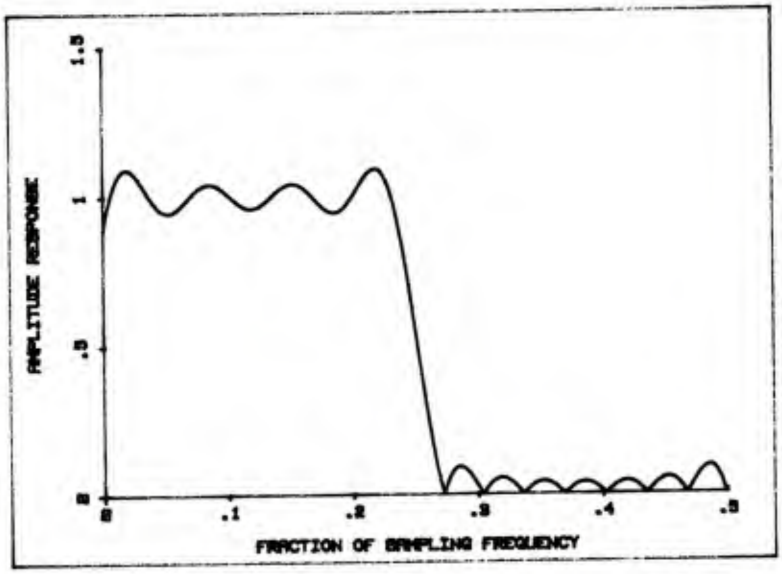

Figure 2. Frequency response of a low pass filter with narrow transition band

NFILT = 33

NBANDS = 2

EDGE(1) = 0

EDGE(2) = .2

EDGE(3) = .3

EDGE(1) = .5

FX(1) = 1.

FX(2) = 0.

WTX(1) = 10.

WTX(2) = 100.

While NFILT is specified as 33, only 17 distinct coefficients are returned, since the design is symmetric about 0. The sample output in Appendix C indicates the ordering of the 17 coefficients, although this is not the filter of Figure 1.

As can be seen in this example, the EDGE array specifies the normalized band edges. The FX array specifies the desired amplitude

response, which is usually (but not necessarily) 1 in the pass bands and 0 in the stop bands. The WTX array specifies a relative scaling of the magnitude of the deviation between the pass band and the stop band. In this example, the design algorithm allows 10 times less deviation in the stop band than in the pass band. This relative weighting may be adjusted arbitrarily to suit a particular need. For example, by relaxing the pass band weighting, say WTX(1)=1., one could design a filter with a more narrow transition band.

For additional information concerning the design of digital filters, the reader is referred to references 1,2, and 3.

V. IMPLEMENTATION OF DIGITAL FILTERS

A digital filter is applied to a data sequence by convoluting the filter weights, or coefficients, with the data points. Specifically, if $\{x_i\}_{i=1}^{n}$ is a sequence of data points equally spaced in time, and if $\{h_k\}_{k=1}^{N}$ are the filter coefficients, where $N \le n$, then the filtered data sequence $\{y_i\}_{i=N}^{n}$ has values given by

$$y_i = \sum_{k=1}^{N} h_k x_{i+1-k} \quad . \tag{1}$$

One notes that if $i \le N-1$, then $i+1-k \le N-k$, so that some subscripts of x in the summation may have values less than 1; we have no corresponding x values. There are two choices: either start the convolution process at i=N, or start at i=1 and modify k to avoid subscripts of x less than 1 until we get to the Nth point. In the first case, N-1 data points at the beginning are unused, and the output sequence starts at i-N. In the second case, the first N-1 output points have not been transformed by the same set of filter coefficients as have the rest of the data; the first N-1 points are of questionable value. It will be noted that the same problem occurs for i>n-N. In what follows, we will discuss a method to avoid these difficulties. In particular, it will be shown that an n-point input sequence can be modified to provide n useful output points.

For nonreal-time applications, i.e.: for post-processing of data, one is in the admirable position of knowing in advance what is going to occur. That is, the convolution process can be numerically manipulated so as to provide one output point corresponding to each input point, with no lag. (Only filters of finite odd length, say N = 2M + 1, M = 1,2,...,63, will be discussed here.) This is accomplished simply by moving the filter coefficients M indeces in Eq. (1), so that

$$y_i = \sum_{k=1}^{2M+1} h_k x_{i-M-1+k} \quad . \tag{2}$$

Eq. (2) implies that the filter coefficients are centered at the i^{th} data

point. If the coefficients are re-indexed as $\left\{h_k'\right\}_{k=-M}^{M}$, then Eq. (2) is more simply written as

$$y_i = \sum_{k=-M}^{M} h_k' x_{i-k} \quad , \tag{3}$$

where $h_k' = h_{k+M+1}$.

Now, for the values $i=1,2,\ldots,M,n-M+1,n-M+2,\ldots,n-1,n$, Eq. (3) still has some values of $i-k$ for which there is no corresponding x.

It is necessary to provide M values at the beginning and M values at the end of the sequence $\left\{x_i\right\}_{i=1}^{n}$. This can be done with a minimum of frequency distortion by using an odd reflection of the first M and last M points. Specifically, for $i-k<1$, define x_{i-k} by

$$x_{i-k} = 2x_1 - x_{k-i+2} \quad . \tag{4}$$

Similary, for $i-k>n$, define x_{i-k} by

$$x_{i-k} = 2x_n - x_{k-i} \quad . \tag{5}$$

Graphically, Eq. (4) reflects $x_2,x_3,\ldots x_M$ about the vertical line through x_1 and then about the horizontal line through x_1. The points x_{n-M+1}, $x_{n-M+2},\ldots x_{n-1}$ are reflected in a like manner about x_n, as shown in Figure 3.

This reflection process can be trivially incorporated into the convolution algorithm, as will be explained below.

The types of digital filters being considered here have an additional property which simplifies the convolution process: they are of either even or odd symmetry about their midpoint. That is,

$$h_k = \pm\, h_{-k} \quad , \quad k=1,2,\ldots M \quad . \tag{6}$$

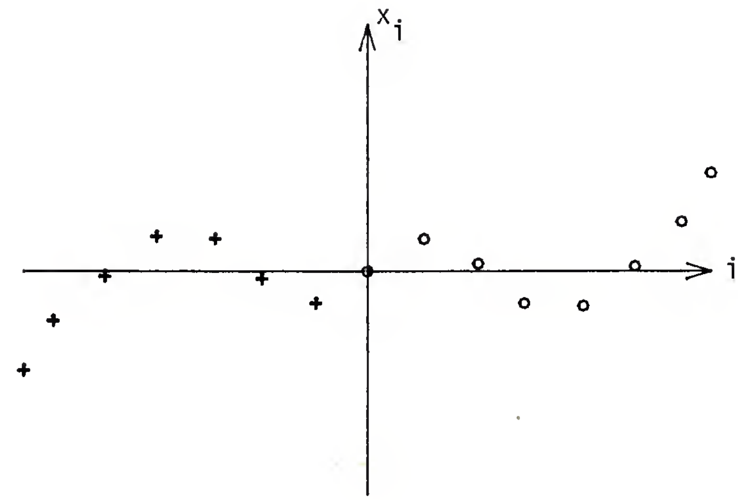

Figure 3. Graphical construction of new end points for x_i

As a consequence, Eq. (3) may be written as

$$y_i = h_o x_i + 2 \sum_{k=1}^{M} h_k (x_{i-k} \pm x_{i+k}) \quad . \tag{7}$$

Whereas Eq. (3) requires 2M + 1 multiplications and 2M additions to implement, Eq. (7) requires only M + 1 multiplications and M additions.

Utilizing Eqs. (4), (5), and (7), the following are three examples of convolution subroutines. The first, in FORTRAN, is in use on the BRL CDC system. The second is in standard BASIC. The third, in an enhanced BASIC, is in use on several BRL systems. In each case, x is the input/output array of length N. The K=M+1 filter coefficients are stored in the array H.

Example 1: FORTRAN Convolution Subroutine

```
SUBROUTINE CONVOL (H,K,X,N)
DIMENSION H(K),X(N),S(127),SAVE(63)
M = K-1
L = K + M
IF(L.GT.127) STOP
DO 5 I=1,M
S(I) = 2.*X(1) - X(K -I)
```

```
      S(L+1-I)=X(I)
      SAVE(I) = 2.*X(N) -X(N-I)
5     CONTINUE
      S(K)=X(1)
      LAST=N-M
      DO 2Ø I=1,N
      X(I) = Ø.
      DO 1Ø J=1,M
      X(I)=X(I)+H(J)*(S(J)+S(L+1-J)
1Ø    CONTINUE
      X(I)=X(I)+H(K)*S(K)
      DO 15 J=2, L
      S(J-1)=S(J)
15    CONTINUE
      1F(I.LE.LAST) S(L)=X(I+K)
      1F(I.GT.LAST) S(L)=SAVE(I-LAST)
2Ø    CONTINUE
      RETURN
      END
```

Example 2: BASIC Convolution Subroutine

```
1ØØ  SUBROUTINE Convolution (H,K,X,N)
11Ø  DIM H(K),X(N),S(127),SAVE(63)
12Ø  M=K-1
13Ø  L=K+M
14Ø  IF L>127 THEN STOP
15Ø  FOR I=1 TO M
16Ø  S(I) = 2.*X(1)-X(K-I)
17Ø  S(L+1-I)=X(I)
18Ø  SAVE(I)=2.*X(N)-X(N-I)
19Ø  NEXT I
2ØØ  S(K)=X(1)
21Ø  Last-N-M
22Ø  FOR I=1 TO N
23Ø  X(I)=Ø
24Ø  FOR J=1 TO M
25Ø  X(I)=X(I)+H(J)*(S(J)+S(L+1-J))
26Ø  NEXT J
27Ø  X(I)=X(I)+H(K)*S(K)
28Ø  FOR J=2 TO L
29Ø  S(J-1=S(J)
3ØØ  NEXT J
31Ø  IF I<=Last THEN S(L)-X(I+K)
32Ø  If I> Last THEN S(L)=SAVE(I-Last)
33Ø  NEXT I
34Ø  RETURN
35Ø  SUBEND
```

In the third example, use is made of several matrix operations available in enhanced BASIC. The function DOT returns the dot product of the two input arrays. The function MAT REORDER rearranges the elements of one array according to the index order specified by another. In this example, the array B has the values 2,3,4,...,L,1, where L=2M+1, the filter length. Implementation of the routine in example 3 represents a decrease in execution time by a factor of 15 over the routine in example 2.

Example 3: BASIC Matrix convolution Subroutine

```
100  SUBROUTINE Convolution(H,L,X,N,B)
110  DIM H(L),X(N),S(127), Save(63),B(L)
120  1F L>127 THEN STOP
130  REDIM S(L)
140  M= INT(L/2)
150  FOR I=1 TO M
160  S(I)=2.*X(1)-X(M+1-I)
170  S(L+1-I)=X(I)
180  Save(I)=2.*X(N)-X(N-I)
190  NEXT I
200  S(M+1)=X(1)
210  Last=N-M
220  FOR I=1 TO N
230  X(I)=DOT(H,S)
240  MAT REORDER S BY B
250  IF I<=Last THEN S(L)=X(I+M+1)
260  IF I>Last THEN S(L)=SAVE(I-Last)
270  NEXT I
280  RETURN
290  SUBEND
```

VI. CONCLUSIONS

Digital filters have a wide range of application for numerical analysis of time-series data. The filter design program presented here has been found to be one of the most versatile available. The reflection principle described in this report seems to introduce the least additional frequency content into the data of any of the methods available. This same technique has been used to produce periodic continuation of essentially transient phenomena, facilitating the use of numerical filters in their analysis.

In a forthcoming BRL Technical Report, the author will discuss specific techniques for the application of digital filters to the analysis of ballistic data. The report will also develop in greater detail the theory and applicability of digital filters to analysis of time series.

VII. SUMMARY

An open literature FORTRAN computer program for the design of finite

impulse-response digital filters has been implemented on the BRL CYBER system. Algorithms have been developed and coded for the convolution of digital filters with time series data. These algorithms include a method for the removal of the filter delay, as well as elimination of the loss of data at the beginning and end of the particular data set being filtered.

VIII. ACKNOWLEDGEMENTS

The author is indebted to Mrs. Emma Wineholt, who made the necessary coding changes in program DESIGN and subroutine FILTER to convert them from IBM to CDC FORTRAN.

REFERENCES

1. J.H. McClellan, T.W. Parks, "A Unified Approach to the Design of Optimum FIR Linear-Phase Digital Filters," IEEE Trans. Circuit Theory, CT-20(6), 697-701 (1973).

2. J.H. McClellan, T.W. Parks, L.R. Rabiner, "A Computer Program for Designing Optimum FIR Linear Phase Digital Filters," IEEE Trans Audio Electroacoustics, AU-21(6), 506-526 (1973).

APPENDIX A

A LISTING OF PROGRAM DESIGN

```
      PROGRAM DESIGN (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7)
C   PROGRAM FOR THE DESIGN OF LINEAR PHASE FINITE IMPULSE
C   RESPONSE (FIR) FILTERS USING THE REMEZ EXCHANGE ALGORITHM
C   JIM MCCLELLAN, RICE UNIVERSITY, APRIL 13, 1973
C   THREE TYPES OF FILTERS ARE INCLUDED--BANDPASS FILTERS,
C   DIFFERENTIATORS, AND HILBERT TRANSFORM FILTERS
C
C   THE INPUT DATA CONSISTS OF 4 CARDS
C
C   CARD 1--FILTER LENGTH, TYPE OF FILTER. 1-MULIPLE
C   PASSBAND/STOPBAND, 2-DIFFERENTIATOR, 3-HILBERT TRANSFORM
C   FILTER.  NUMBER OF BANDS, CARD PUNCH DESIRED, AND GRID
C   DENSITY.
C
C   CARD 2--BANDEDGES.  LOWER AND UPPER EDGES FOR EACH BAND
C   WITH A MAXIMUM OF 10 BANDS.
C
C   CARD 3--DESIRED FUNCTION (OR DESIRED SLOPE IF A
C   DIFFERENTIATOR) FOR EACH BAND.
C
C   CARD 4--WEIGHT FUNCTION IN EACH BAND.  FOR A
C   DIFFERENIATOR, THE WEIGHT FUNCTION IS INVERSELY
C   PROPORTIONAL TO F.
C
C   THE FOLLOWING INPUT DATA SPECIFIES A LENGTH 32 BANDPASS
C   FILTER WITH STOPBANDS 0 TO 0.1 AND 0.425 TO 0.5, AND
C   PASSBAND FROM 0.2 TO 0.35 WITH WEIGHTING OF 10 IN THE
C   STOPBANDS AND 1 IN THE PASSBAND. THE IMPULSE RESPONSE
C   WILL BE PUNCHED AND THE GRID DENSITY IS 32. THIS IS THE
C   FILTER IN FIGURES 9 AND 10 IN THE TEXT.
C   SAMPLE INPUT DATA SETUP
C   32,1,2,1,32
C   0,0.1,0.2,0.35,0.425,0.5
C   0,1,0
C   10,1,10
```

```
C
C  THE FOLLOWING INPUT DATA SPECIFIES A LENGTH 32 WIDEBAND
C  DIFFERENTIATOR WITH SLOPE 1 AND WEIGHING OF 1/P. THE
C  IMPULSE RESPONSE WILL NOT BE PUNCHED AND THE GRID
C  DENSITY IS ASSUMED TO BE 16. THIS IS THE FILTER IN
C  FIGURES 17 AND 18 IN THE TEXT.
C  32,2,1,0,0
C  0,0.5
C  1.0
C  1.0
C
C
      COMMON PI2,AD,DEV,X,Y,GRID,DES,WT,ALPHA,IEXT,NFCNS,NGRID
      DIMENSION IEXT(66),AD(66),ALPHA(66),X(66),Y(66)
      DIMENSION H(66)
      DIMENSION DES(1045),GRID(1045),WT(1045)
      DIMENSION EDGE(20),FX(10),WTX(10),DEVIAT(10)
      DOUBLE PRECISION AD,DEV,X,Y
      DOUBLE PRECISION PI2,PI
      PI2=6.283185307179586D0
      PI=3.141592653589793D0
C
C  THIS PROGRAM IS SET UP FOR A MAXIMUM LENGTH OF 128, BUT
C  THIS UPPER LIMIT CAN BE CHANGE BY REDIMENSIONING THE
C  ARRAYS IEXT,AD,ALPHA,X,Y,H TO BE NFMAX/2 + 2.
C  THE ARRAYS DES, GRID, AND WT MUST BE DIMENSIONED
C  16(NFMAX/2+2).
C
      NFMAX=128
  100 CONTINUE
      JTYPE=0
C
C  PROGRAM INPUT SECTION
C
      READ(5,1000) NFILT,JTYPE,NBANDS,JPUNCH,LGRID
```

```
 1000 FORMAT(I4,I1,I2,I1,I4)
      IF(EOF(5).NE.0) GOTO 700
      IF(NFILT.GT.NFMAX.OR.NFILT.LT.3) CALL ERROR
      IF(NBANDS.LE.0) NBANDS=1
C
C  GRID DENSITY IS ASSUMED TO BE 16 UNLESS SPECIFIED
C  OTHERWISE
C
      IF(LGRID.LE.0) LGRID=16
      JB=2*NBANDS
      READ(5,1010) (EDGE(J),J=1,JB)
 1010 FORMAT(10F8.0)
      READ(5,1020) (FX(J),J=1,NBANDS)
 1020 FORMAT(10F8.0)
      READ(5,1030) (WTX(J),J=1,NBANDS)
 1030 FORMAT(10F8.0)
      IF(JTYPE.EQ.0) CALL ERROR
      NEG=1
      IF(JTYPE.EQ.1) NEG=0
      NODD=NFILT/2
      NODD=NFILT-2*NODD
      NFCNS=NFILT/2
      IF(NODD.EQ.1.AND.NEG.EQ.0) NFCNS=NFCNS+1
      IF((LGRID*NFCNS).GT.(16*(NFMAX/2+2)))  CALL ERROR
C
C  SET UP THE DENSE GRID.  THE NUMBER OF POINTS IN THE GRID
C  IS (FILTER LENGTH + 1)*GRID DENSITY/2
C
      GRID(1)=EDGE(1)
      DELF=LGRID*NFCNS
      DELF=0.5/DELF
      IF(NEG.EQ.0) GO TO 135
      IF(EDGE(1).LT.DELF) GRID(1)=DELF
  135 CONTINUE
      J=1
```

```
      L=1
      LBAND=1
  140 FUP=EDGE(L+1)
  145 TEMP=GRID(J)
C
C  CALCULATE THE DESIRED MAGNITUDE RESPONSE AND THE WEIGHT
C  FUNCTION OF THE GRID.
C
      DES(J)=EFF(TEMP,FX,WTX,LBAND,JTYPE)
      WT(J)=WATE(TEMP,FX,WTX,LBAND,JTYPE)
      J=J+1
      GRID(J)=TEMP+DELF
      IF(GRID(J).GT.FUP) GO TO 150
      GO TO 145
  150 GRID(J-1)=FUP
      DES(J-1)=EFF(FUP,FX,WTX,LBAND,JTYPE)
      WT(J-1)=WATE(FUP,FX,WTX,LBAND,JTYPE)
      LBAND=LBAND+1
      L=L+2
      IF(LBAND.GT.NBANDS) GO TO 160
      GRID(J)=EDGE(L)
      GO TO 140
  160 NGRID=J-1
      IF(NEG.NE.NODD) GO TO 165
      IF(GRID(NGRID).GT.(0.5-DELF)) NGRID=NGRID-1
  165 CONTINUE
C
C  SET UP THE NEW APPROXIMATION PROBLEM WHICH IS EQUIVALENT
C  TO THE ORIGINAL PROBLEM
C
      IF(NEG) 170,170,180
  170 IF(NODD.EQ.1) GO TO 200
      DO 175 J=1,NGRID
      CHANGE=DCOS(PI*GRID(J))
      DES(J)=DES(J)/CHANGE
```

```
  175 WT(J)=WT(J)*CHANGE
      GO TO 200
  180 IF(NODD.EQ.1) GO TO 190
      DO 185 J=1,NGRID
      CHANGE=DSIN(PI*GRID(J))
      DES(J)=DES(J)/CHANGE
  185 WT(J)=WT(J)*CHANGE
      GO TO 200
  190 DO 195 J=1,NGRID
      CHANGE=DSIN(PI2*GRID(J))
      DES(J)=DES(J)/CHANGE
  195 WT(J)=WT(J)*CHANGE
C
C  INITIAL GUESS FOR THE EXTREMAL FREQUENCIES--EQUALLY
C  SPACED ALONG THE GRID
C
  200 TEMP=FLOAT(NGRID-1)/FLOAT(NFCNS)
      DO 210 J=1,NFCNS
  210 IEXT(J)=(J-1)*TEMP+1
      IEXT(NFCNS+1)=NGRID
      NM1=NFCNS-1
      NZ=NFCNS+1
C
C  CALL REMEZ EXCHANGE ALGORITHIM TO DO THE APPROXIMATION
C  PROBLEM
C
      CALL REMEZ(EDGE,NBANDS)
C
C  CALCULATE THE IMPULSE RESPONSE
C
      IF(NEG) 300,300,320
  300 IF(NODD.EQ.0) GO TO 310
      DO 305 J=1,NM1
  305 H(J)=0.5*ALPHA(NZ-J)
      H(NFCNS)=ALPHA(1)
```

```
      GO TO 350
  310 H(1)=0.25*ALPHA(NFCNS)
      DO 315 J=2,NM1
  315 H(J)=0.25*(ALPHA(NZ-J)+ALPHA(NFCNS+2-J))
      H(NFCNS)=0.5*ALPHA(1)+0.25*ALPHA(2)
      GO TO 350
  320 IF(NODD.EQ.0) GO TO 330
      H(1)=0.25*ALPHA(NFCNS)
      H(2)=0.25*ALPHA(NM1)
      DO 325 J=3,NM1
  325 H(J)=0.25*(ALPHA(NZ-J)-ALPHA(NFCNS+3-J))
      H(NFCNS)=0.5*ALPHA(1)-0.25*ALPHA(3)
      H(NZ)=0.0
      GO TO 350
  330 H(1)=0.25*ALPHA(NFCNS)
      DO 335 J=2,NM1
  335 H(J)=0.25*(ALPHA(NZ-J)-ALPHA(NFCNS+2-J))
      H(NFCNS)=0.5*ALPHA(1)-0.25*ALPHA(2)
C
C  PROGRAM OUTPUT SECTION
C
  350 PRINT 360
  360 FORMAT(1H1, 70(1H*)//25X,'FINITE IMPULSE RESPONSE (FIR)'/
     125X,'LINEAR PHASE DIGITAL FILTER DESIGN'/
     225X,'REMEZ EXCHANGE ALGORITHIM'/)
      IF(JTYPE.EQ.1) PRINT 365
  365 FORMAT(25X,'BANDPASS FILTER'/)
      IF(JTYPE.EQ.2) PRINT 370
  370 FORMAT(25X,'DIFFERENTIATOR'/)
      IF(JTYPE.EQ.3) PRINT 375
  375 FORMAT(25X,'HILBERT TRANSFORMER'/)
      PRINT 378,NFILT
  378 FORMAT(15X,'FILTER LENGTH=',I3/)
      PRINT 380
  380 FORMAT(15X,'***** IMPULSE RESPONSE *****')
```

```
      DO 381 J=1,NFCNS
      K=NFILT+1-J
      IF(NEG.EQ.0) PRINT 382,J,H(J),K
      IF(NEG.EQ.1)  PRINT 383,J,H(J),K
  381 CONTINUE
  382 FORMAT(20X,'H(',I3,')=',E15.8,' = H(',I4,')')
  383 FORMAT(20X,'H(',I3,')=',E15.8,' = -H(',I4,')')
      IF(NEG.EQ.1.AND.NODD.EQ.1) PRINT 384,NZ
  384 FORMAT(20X,'H(',I3,') = 0.0')
      DO 450 K=1,NBANDS,4
      KUP=K+3
      IF(KUP.GT.NBANDS) KUP=NBANDS
      PRINT 385,(J,J=K,KUP)
  385 FORMAT(/24X,4('BAND',I3,8X))
      PRINT 390,(EDGE(2*J-1),J=K,KUP)
  390 FORMAT(2X,'LOWER BAND EDGE',5F15.9)
      PRINT 395,(EDGE(2*J),J=K,KUP)
  395 FORMAT (2X,'UPPER BAND EDGE',5F15.9)
      IF(JTYPE.NE.2) PRINT 400,(FX(J),J=K,KUP)
  400 FORMAT(2X,'DESIRED VALUE',2X,5F15.9)
      IF(JTYPE.EQ.2) PRINT 405,(FX(J),J=K,KUP)
  405 FORMAT(2X,'DESIRED SLOPE',2X,5F15.9)
      PRINT 410,(WTX(J),J=K,KUP)
  410 FORMAT(2X,'WEIGHTING',6X,5F15.9)
      DO 420 J=K,KUP
  420 DEVIAT(J)=DEV/WTX(J)
      PRINT 425,(DEVIAT(J),J=K,KUP)
  425 FORMAT(2X,'DEVIATION',6X,5F15.9)
      IF(JTYPE.NE.1) GO TO 450
      DO 430 J=K,KUP
  430 DEVIAT(J)=20.0*ALOG10(DEVIAT(J))
      PRINT 435,(DEVIAT(J),J=K,KUP)
  435 FORMAT(2X,'DEVIATION IN DB',5D15.9)
  450 CONTINUE
      PRINT 455,(GRID(IEXT(J)),J=1,NZ)
```

```
 455 FORMAT(/2X,'EXTREMAL FREQUENCIES'/(2X,5F12.7))
     PRINT 460
 460 FORMAT(/1X,70(1H*)/1H1)
     IF(JPUNCH.NE.0) WRITE(7,2000) (H(J),J=1,NFCNS)
2000 FORMAT(5E15.8)
     IF(NFILT.NE.0) GO TO 100
 700 STOP
     END
```

```
      SUBROUTINE REMEZ(EDGE,NBANDS)
C
C  THIS SUBROUTINE IMPLEMENTS THE REMEZ EXCHANGE ALGORITHM
C  FOR THE WEIGHTED CHEBYCHEV APPROXIMATION OF A CONTINUOUS
C  FUNCTION WITH A SUM OF COSINES.  INPUTS TO THE SUBROUTINE
C  ARE A DENSE GRID  WHICH REPLACES THE FREQUENCY AXIS, THE
C  DESIRED FUNCTION ON THIS GRID, THE WEIGHT FUNCTION ON THE
C  GRID, THE NUMBER OF COSINES, AND AN INITIAL GUESS OF THE
C  EXTERNAL FREQUENCIES.  THE PROGRAM MINIMIZES THE CHEBYCHEV
C  ERROR BY DETERMINING THE BEST LOCATION OF THE EXTERNAL
C  FREQUENCIES (POINTS OF MAXIMUM ERROR) AND THEN CALCULATES
C  THE COEFFICIENTS OF THE BEST APPROXIMATION.
C
      COMMON P12,AD,DEV,X,Y,GRID,DES,WT,ALPHA,IEXT,NFCNS,NGRID
       DIMENSION EDGE(20)
      DIMENSION IEXT(66),AD(66),ALPHA(66),X(66),Y(66)
      DIMENSION DES(1045),GRID(1045),WT(1045)
      DIMENSION A(66),P(65),Q(65)
      DOUBLE PRECISION P12,DNUM,DDEN,DTEMP,A,P,Q
      DOUBLE PRECISION AD,DEV,X,Y
C
C  THIS PROGRAM ALLOWS A MAXIMUM NUMBER OF ITERATIONS OF 25
C
      ITRMAX=25
      DEVL=-1.0
      NZ=NFCNS+1
      NZZ=NFCNS+2
      NITER=0
  100 CONTINUE
      IEXT(NZZ)=NGRID+1
      NITER=NITER+1
      IF(NITER.GT.ITRMAX) GO TO 400
      DO 110 J=1,NZ
      DTEMP=GRID(IEXT(J))
      DTEMP=DCOS(DTEMP*P12)
```

```
  110 X(J)=DTEMP
      JET=(NFCNS-1)/15+1
      DO 120 J=1,NZ
  120 AD(J)=D(J,NZ,JET)
      DNUM=0.0
      DDEN=0.0
      K=1
      DO 130 J=1,NZ
      L=IEXT(J)
      DTEMP=AD(J)*DES(L)
      DNUM=DNUM+DTEMP
      DTEMP=K*AD(J)/WT(L)
      DDEN=DDEN+DTEMP
  130 K=-K
      DEV=DNUM/DDEN
      NU=1
      IF(DEV.GT.0.0) NU=-1
      DEV=-NU*DEV
      K=NU
      DO 140 J=1,NZ
      L=IEXT(J)
      DTEMP=K*DEV/WT(L)
      Y(J)=DES(L)+DTEMP
  140 K=-K
      IF(DEV.GE.DEVL) GO TO 150
      CALL OUCH
      GO TO 400
  150 DEVL=DEV
      JCHNGE=0
      K1=IEXT(1)
      KNZ=IEXT(NZ)
      KLOW=0
      NUT=-NU
      J=1
```

```
C  SEARCH FOR THE EXTERNAL FREQUENCIES OF THE BEST
C  APPROXIMATIONS
C
  200 IF(J.EQ.NZZ) YNZ=COMP
      IF(J.GE.NZZ) GO TO 300
      KUP=IEXT(J+1)
      L=IEXT(J)+1
      NUT=-NUT
      IF(J.EQ.2) Y1=COMP
      COMP=DEV
      IF(L.GE.KUP) GO TO 220
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 220
      COMP=NUT*ERR
  210 L=L+1
      IF(L.GE.KUP) GO TO 215
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 215
      COMP=NUT*ERR
      GO TO 210
  215 IEXT(J)=L-1
      J=J+1
      KLOW=L-1
      JCHNGE=JCHNGE+1
      GO TO 200
  220 L=L-1
  225 L=L-1
      IF(L.LE.KLOW) GO TO 250
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
```

```
      IF(DTEMP.GT.0.0) GO TO 230
      IF(JCHNGE.LE.0) GO TO 225
      GO TO 260
  230 COMP=NUT*ERR
  235 L=L-1
      IF(L.LE.KLOW) GO TO 240
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0)GO TO 240
      COMP=NUT*ERR
      GO TO 235
  240 KLOW=IEXT(J)
      IEXT(J)=L+1
      J=J+1
      JCHNGE=JCHNGE+1
      GO TO 200
  250 L=IEXT(J)+1
      IF(JCHNGE.GT.0) GO TO 215
  255 L=L+1
      IF(L.GE.KUP) GO TO 260
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 255
      COMP=NUT*ERR
      GO TO 210
  260 KLOW=IEXT(J)
      J=J+1
      GO TO 200
  300 IF(J.GT.NZZ) GO TO 320
      IF(K1.GT.IEXT(1)) K1=IEXT(1)
      IF(KNZ.LT.IEXT(NZ)) KNZ=IEXT(NZ)
      NUT1=NUT
      NUT=-NU
```

```
      L=0
      KUP=K1
      COMP=YNZ*(1.00001)
      LUCK=1
  310 L=L+1
      IF(L.GE.KUP) GO TO 315
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 310
      COMP=NUT*ERR
      J=NZZ
      GO TO 210
  315 LUCK=6
      GO TO 325
  320 IF(LUCK.GT.9) GO TO 350
      IF(COMP.GT.Y1) Y1=COMP
      K1=IEXT(NZZ)
  325 L=NGRID+1
      KLOW=KNZ
      NUT=-NUT1
      COMP=Y1*(1.00001)
  330 L=L-1
      IF(L.LE.KLOW) GO TO 340
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 330
      J=NZZ
      COMP=NUT*ERR
      LUCK=LUCK+10
      GO TO 235
  340 IF(LUCK.EQ.6) GO TO 370
      DO 345 J=1,NFCNS
  345 IEXT(NZZ-J)=IEXT(NZ-J)
```

```
      IEXT(1)=K1
      GO TO 100
  350 KN=IEXT(NZZ)
      DO 360 J=1,NFCNS
  360 IEXT(J)=IEXT(J+1)
      IEXT(NZ)=KN
      GO TO 100
  370 IF(JCHNGE.GT.0) GO TO 100
C
C  CALCULATION OF THE COEFFICIENTS OF THE BEST APPROXIMATION
C  USING THE INVERSE DISCRETE FOURIER TRANSFORM
C
  400 CONTINUE
      NM1=NFCNS-1
      FSH=1.0E-06
      GTEMP=GRID(1)
      X(NZZ)=-2.0
      CN=2*NFCNS-1
      DELF=1.0/CN
      L=1
      KKK=0
      IF (EDGE(1) .EQ. 0.  .AND.  EDGE(2*NBANDS) .EQ. 0.5 ) KKK=1
      IF(NFCNS.LE.3) KKK=1
      IF(KKK.EQ.1) GO TO 405
      DTEMP=DCOS(PI2*GRID(1))
      DNUM=DCOS(PI2*GRID(NGRID))
      AA=2.0/(DTEMP-DNUM)
      BB=-(DTEMP+DNUM)/(DTEMP-DNUM)
  405 CONTINUE
      DO 430 J=1,NFCNS
      FT=(J-1)*DELF
      XT=DCOS(PI2*FT)
      IF(KKK.EQ.1) GO TO 410
      XT=(XT-BB)/AA
      FT=ARCOS(XT)/PI2
```

```
410 XE=X(L)
    IF(XT.GT.XE) GO TO 420
    IF((XE-XT).LT.FSH) GO TO 415
    L=L+1
    GO TO 410
415 A(J)=Y(L)
    GO TO 425
420 IF((XT-XE).LT.FSH) GO TO 415
    GRID(1)=FT
    A(J)=GEE(1,NZ)
425 CONTINUE
    IF(L.GT.1) L=L-1
430 CONTINUE
    GRID(1)=GTEMP
    DDEN=PI2/CN
    DO 510 J=1,NFCNS
    DTEMP=0.0
    DNUM=(J-1)*DDEN
    IF(NM1.LT.1) GO TO 505
    DO 500 K=1,NM1
500 DTEMP=DTEMP+A(K+1)*DCOS(DNUM*K)
505 DTEMP=2.0*DTEMP+A(1)
510 ALPHA(J)=DTEMP
    DO 550 J=2,NFCNS
550 ALPHA(J)=2*ALPHA(J)/CN
    ALPHA(1)=ALPHA(1)/CN
    IF(KKK.EQ.1) GO TO 545
    P(1)=2.0*ALPHA(NFCNS)*BB+ALPHA(NM1)
    P(2)=2.0*AA*ALPHA(NFCNS)
    Q(1)=ALPHA(NFCNS-2)-ALPHA(NFCNS)
    DO 540 J=2,NM1
    IF(J.LT.NM1) GO TO 515
    AA=0.5*AA
    BB=0.5*BB
515 CONTINUE
```

```
      P(J+1)=0.0
      DO 520 K=1,J
      A(K)=P(K)
  520 P(K)=2.0*BB*A(K)
      P(2)=P(2)+A(1)*2.0*AA
      JM1=J-1
      DO 525 K=1,JM1
  525 P(K)=P(K)+Q(K)+AA*A(K+1)
      JP1=J+1
      DO 530 K=3,JP1
  530 P(K)=P(K)+AA*A(K-1)
      IF(J.EQ.NM1) GO TO 540
      DO 535 K=1,J
  535 Q(K)=-A(K)
      Q(1)=Q(1)+ALPHA(NFCNS-1-J)
  540 CONTINUE
      DO 543 J=1,NFCNS
  543 ALPHA(J)=P(J)
  545 CONTINUE
      IF(NFCNS.GT.3) RETURN
      ALPHA(NFCNS+1)=0.0
      ALPHA(NFCNS+2)=0.0
      RETURN
      END
```

```
      FUNCTION WATE(TEMP,FX,WTX,LBAND,JTYPE)
C
C   FUNCTION TO CALCULATE THE WEIGHT FUNCTION AS A FUNCTION
C   OF FREQUENCY.
C
      DIMENSION FX(5),WTX(5)
      IF(JTYPE.EQ.2) GO TO 1
      WATE=WTX(LBAND)
      RETURN
    1 IF(FX(LBAND).LT.0.0001) GO TO 2
      WATE=WTX(LBAND)/TEMP
      RETURN
    2 WATE=WTX(LBAND)
      RETURN
      END
```

```
      SUBROUTINE ERROR
      PRINT 1
    1 FORMAT(' ************ ERROR IN INPUT DATA ************ ')
      STOP
      END
```

```
      FUNCTION EFF(TEMP,FX,WTX,LBAND,JTYPE)
C
C  FUNCTION TO CALCULATE THE DESIRED RESPONSE MAGNITUDE
C  AS A FUNCTION OF FREQUENCY.
C
      DIMENSION FX(5),WTX(5)
      IF(JTYPE.EQ.2) GO TO 1
      EFF=FX(LBAND)
      RETURN
    1 EFF=FX(LBAND)*TEMP
      RETURN
      END
```

```
      SUBROUTINE OUCH
      PRINT 1
    1 FORMAT(' ************ FAILURE TO CONVERGE ************'/
     1'0PROBABLE CAUSE IS MACHINE ROUNDING ERROR'/
     2'0THE IMPULSE RESPONSE MAY BE CORRECT'/
     3'0CHECK WITH A FREQUENCY RESPONSE')
      RETURN
       END
      DOUBLE PRECISION FUNCTION GEE(K,N)
C
C  FUNCTION TO EVALUATE THE FREQUENCY RESPONSE USING THE
C  LAGRANGE INTERPOLATION FORMULA IN THE BARYCENTRIC FORM
C
      COMMON P12,AD,DEV,X,Y,GRID,DES,WT,ALPHA,IEXT,NFCNS,NGRID
      DIMENSION IEXT(66),AD(66),ALPHA(66),X(66),Y(66)
      DIMENSION DES(1045),GRID(1045),WT(1045)
      DOUBLE PRECISION P,C,D,XF
      DOUBLE PRECISION P12
      DOUBLE PRECISION AD,DEV,X,Y
      P=0.0
      XF=GRID(K)
      XF=DCOS(P12*XF)
      D=0.0
      DO 1 J=1,N
      C=XF-X(J)
      C=AD(J)/C
      D=D+C
    1 P=P+C*Y(J)
      GEE=P/D
      RETURN
      END
```

```
      DOUBLE PRECISION FUNCTION D(K,N,M)
C
C  FUNCTION TO CALCULATE THE LAGRANGE INTERPOLATION
C  COEFFICIENTS FOR USE IN THE FUNCTION GEE.
C
      COMMON P12,AD,DEV,X,Y,GRID,DES,WT,ALPHA,IEXT,NFCNS,NGRID
      DIMENSION IEXT(66),AD(66),ALPHA(66),X(66),Y(66)
      DIMENSION DES(1045),GRID(1045),WT(1045)
      DOUBLE PRECISION AD,DEV,X,Y
      DOUBLE PRECISION Q
      DOUBLE PRECISION P12
      D=1.0
      Q=X(K)
      DO 3 L=1,M
      DO 2 J=L,N,M
      IF(J-K)1,2,1
    1 D=2.0*D*(Q-X(J))
    2 CONTINUE
    3 CONTINUE
      D=1.0/D
      RETURN
      END
```

APPENDIX B

A LISTING OF SUBROUTINE FILTER

```
      SUBROUTINE FILTER(NFILT,NBANDS,EDGE,FX,WTX,IPRINT,H)
C ******************************************************************
C
C SUBROUTINE FILTER - PGMR. JAMES N. WALBERT, NOVEMBER 1974
C
C DESIGNS A DIGITAL FILTER OF UP TO 10 BANDS AND OF MAXIMUM LENGTH 127.
C THE SUBROUTINE USES THE REMEZ EXCHANGE ALGORITHM TO FIND THE BEST
C APPROXIMATION WHICH MINIMIZES CHEBYSHEV ERROR.
C
C USAGE -
C
C       CALL FILTER(NFILT,NBANDS,EDGE,FX,WTX,IPRINT,H)
C
C DESCRIPTION OF VARIABLES -
C
C       NFILT   - FILTER LENGTH. MUST BE AN ODD NUMBER BETWEEN 3
C                 AND 127, INCLUSIVE.
C       NBANDS  - NUMBER OF PASS-STOP BANDS. MAXIMUM IS 10.
C       EDGE    - ARRAY DEFINING BAND EDGES. DIMENSION IN CALLING
C                 PROGRAM MUST BE 2*NBANDS. EDGE IS DEFINED BY
C                 FREQUENCY/SAMPLING RATE. EDGE(1)=0.,
C                 EDGE(2*NBANDS)=.5.
C       FX      - DESIRED FUNCTION. FX IS 1. IN PASSBANDS AND 0. IN
C                 STOPBANDS.
C       WTX     - WEIGHTING FACTOR. USUALLY 10. IN PASSBANDS AND
C                 100. IN STOPBANDS.
C       IPRINT  - CONTROL VARIABLE. IF IPRINT=0,COEFFICIENTS ARE
C                 PRINTED. IF IPRINT=1,COEFFICIENTS ARE NOT PRINTED.
C       H       - FILTER COEFFICIENT ARRAY. NUMBERED FROM 1 TO
C                 NFILT/2+1. THE ARRAY IS SYMMETRIC ABOUT H(NFILT/2+1).
C                 THE H ARRAY SHOULD HAVE DIMENSION NFILT IN THE
C                 CALLING PROGRAM.
C
C FUNCTION SUBPROGRAMS CALLED -
C
```

```
C        DOUBLE PRECISION FUNCTION GEE
C
C
C
C
C ***********************************************************************
C
      DIMENSION IEXT(66),AD(66),ALPHA(66),X(66),Y(66)
      DIMENSION DEVIAT(10),AMP(66),FRE(66),FXA(20)
      COMMON /XFRBLK/ DES(1045),GRID(1045),WT(1045),A(66)
      DIMENSION EDGE(1),FX(1),WTX(1),H(1)
      DOUBLE PRECISION AD,DEV,X,Y,PI2,PI
      DOUBLE PRECISION DNUM,DDEN,DTEMP,A
      PI2=6.283185307179586
      PI=3.141592653589793
      NFCNS=NFILT/2+1
      GRID(1)=EDGE(1)
      DELF=16*NFCNS
      DELF=0.5/DELF
      J=1
      L=1
      LBAND=1
140   FUP=EDGE(L+1)
145   TEMP=GRID(J)
      DES(J)=FX(LBAND)
      WT(J)=WTX(LBAND)
      J=J+1
      GRID(J)=TEMP+DELF
      IF(GRID(J).GT.FUP) GO TO 150
      GO TO 145
150   GRID(J-1)=FUP
      DES(J-1)=FX(LBAND)
      WT(J-1)=WTX(LBAND)
      LBAND=LBAND+1
      L=L+2
```

```
      IF(LBAND.GT.NBANDS) GO TO 160
      GRID(J)=EDGE(L)
      GO TO 140
160   NGRID=J-1
200   TEMP=FLOAT(NGRID-1)/FLOAT(NFCNS)
      DO 210 J=1,NFCNS
210   IEXT(J)=(J-1)*TEMP+1
      IEXT(NFCNS+1)=NGRID
      NM1=NFCNS-1
      NZ=NFCNS+1
      DEVL=-1.
      NZZ=NFCNS+2
      NITER=0
1000  CONTINUE
      IEXT(NZZ)=NGRID+1
      NITER=NITER+1
      IF(NITER.GT.25) GO TO 4000
      DO 1100 J=1,NZ
      DTEMP=GRID(IEXT(J))
      DTEMP=DCOS(DTEMP*PI2)
1100  X(J)=DTEMP
      JET=(NFCNS-1)/15+1
      DO 1200 J=1,NZ
      D=1.
      DO 1193 LL=1,JET
      DO 1192 KK=LL,NZ,JET
      IF(KK-J)1191,1192,1191
1191  D=2.0*D*(X(J)-X(KK))
1192  CONTINUE
1193  CONTINUE
1200  AD(J)=1.0/D
      DNUM=0.0
      DDEN=0.0
      K=1
      DO 1300 J=1,NZ
```

```
      L=IEXT(J)
      DTEMP=AD(J)*DES(L)
      DNUM=DNUM+DTEMP
      DTEMP=K*AD(J)/WT(L)
      DDEN=DDEN+DTEMP
1300  K=-K
      DEV=DNUM/DDEN
      NU=1
      IF(DEV.GT.0.0) NU=-1
      DEV=-NU*DEV
      K=NU
      DO 1400 J=1,NZ
      L=IEXT(J)
      DTEMP=K*DEV/WT(L)
      Y(J)=DES(L)+DTEMP
1400  K=-K
      IF(DEV.GE.DEVL) GO TO 1500
      PRINT 1401
1401  FORMAT(1H0,'**** FAILURE TO CONVERGE ****'/'0RESPONSE MAY BE OK')
      GO TO 4000
1500  DEVL=DEV
      JCHNGE=0
      K1=IEXT(1)
      KNZ=IEXT(NZ)
      KLOW=0
      NUT=-NU
      J=1
2000  IF(J.EQ.NZZ) YNZ=COMP
      IF(J.GE.NZZ) GO TO 3000
      KUP=IEXT(J+1)
      L=IEXT(J)+1
      NUT=-NUT
      IF(J.EQ.2) Y1=COMP
      COMP=DEV
      IF(L.GE.KUP) GO TO 2200
```

```
      ERR=GEE(NZ,GRID(L),X,AD,Y)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 2200
      COMP=NUT*ERR
2100  L=L+1
      IF(L.GE.KUP) GO TO 2150
      ERR=GEE(NZ,GRID(L),X,AD,Y)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 2150
      COMP=NUT*ERR
      GO TO 2100
2150  IEXT(J)=L-1
      J=J+1
      KLOW=L-1
      JCHNGE=JCHNGE+1
      GO TO 2000
2200  L=L-1
2250  L=L-1
      IF(L.LE.KLOW) GO TO 2500
      ERR=GEE(NZ,GRID(L),X,AD,Y)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.GT.0.0) GO TO 2300
      IF(JCHNGE.LE.0) GO TO 2250
      GO TO 2600
2300  COMP=NUT*ERR
2350  L=L-1
      IF(L.LE.KLOW) GO TO 2400
      ERR=GEE(NZ,GRID(L),X,AD,Y)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 2400
      COMP=NUT*ERR
```

```
      GO TO 2350
 2400 KLOW=IEXT(J)
      IEXT(J)=L+1
      J=J+1
      JCHNGE=JCHNGE+1
      GO TO 2000
 2500 L=IEXT(J)+1
      IF(JCHNGE.GT.0) GO TO 2150
 2550 L=L+1
      IF(L.GE.KUP) GO TO 2600
      ERR=GEE(NZ,GRID(L),X,AD,Y)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 2550
      COMP=NUT*ERR
      GO TO 2100
 2600 KLOW=IEXT(J)
      J=J+1
      GO TO 2000
 3000 IF(J.GT.NZZ) GO TO 3200
      IF(K1.GT.IEXT(1)) K1=IEXT(1)
      IF(KNZ.LT.IEXT(NZ)) KNZ=IEXT(NZ)
      NUT1=NUT
      NUT=-NU
      L=0
      KUP=K1
      COMP=YNZ*(1.00001)
      LUCK=1
 3100 L=L+1
      IF(L.GE.KUP) GO TO 3150
      ERR=GEE(NZ,GRID(L),X,AD,Y)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 3100
      COMP=NUT*ERR
```

```
      J=NZZ
      GO TO 2100
3150  LUCK=6
      GO TO 3250
3200  IF(LUCK.GT.9) GO TO 3500
      IF(COMP.GT.Y1) Y1=COMP
      K1=IEXT(NZZ)
3250  L=NGRID+1
      KLOW=KNZ
      NUT=-NUT1
      COMP=Y1*(1.00001)
3300  L=L-1
      IF(L.LE.KLOW) GO TO 3400
      ERR=GEE(NZ,GRID(L),X,AD,Y)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.) GO TO 3300
      J=NZZ
      COMP=NUT*ERR
      LUCK=LUCK+10
      GO TO 2350
3400  IF(LUCK.EQ.6) GO TO 3700
      DO 3450 J=1,NFCNS
3450  IEXT(NZZ-J)=IEXT(NZ-J)
      IEXT(1)=K1
      GO TO 1000
3500  KN=IEXT(NZZ)
      DO 3600 J=1,NFCNS
3600  IEXT(J)=IEXT(J+1)
      IEXT(NZ)=KN
      GO TO 1000
3700  IF(JCHNGE.GT.0) GO TO 1000
4000  CONTINUE
      NM1=NFCNS-1
      FSH=1.0E-06
```

```
      X(NZZ)=-2.0
      CN=2*NFCNS-1
      DELF=1.0/CN
      L=1
      DO 4300 J=1,NFCNS
      FT=(J-1)*DELF
      XT=DCOS(P12*FT)
4100  XE=X(L)
      IF(XT.GT.XE) GO TO 4200
      IF((XE-XT).LT.FSH) GO TO 4150
      L=L+1
      GO TO 4100
4150  A(J)=Y(L)
       GO TO 4250
4200  IF((XT-XE).LT.FSH) GO TO 4150
      A(J)=GEE(NZ,FT,X,AD,Y)
4250  CONTINUE
      IF(L.GT.1) L=L-1
4300  CONTINUE
      DDEN=P12/CN
      DO 5100 J=1,NFCNS
      DTEMP=0.
      DNUM=(J-1)*DDEN
      IF(NM1.LT.1) GO TO 5050
      DO 5000 K=1,NM1
5000  DTEMP=DTEMP+A(K+1)*DCOS(DNUM*FLOAT(K))
5050  DTEMP=2.0*DTEMP+A(1)
5100  ALPHA(J)=DTEMP
      DO 5500 J=2,NFCNS
5500  ALPHA(J)=2*ALPHA(J)/CN
      ALPHA(1)=ALPHA(1)/CN
      IF(NFCNS.GT.3) GO TO 304
      ALPHA(NFCNS+1)=0.
      ALPHA(NFCNS+2)=0.
304   CONTINUE
```

```
      DO 305 J=1,NM1
305   H(J)=0.5*ALPHA(NZ-J)
      H(NFCNS)=ALPHA(1)
      IF(IPRINT.EQ.1) GO TO 700
350   PRINT 360
  360 FORMAT(1H1, 70(1H*)//25X,'FINITE IMPULSE RESPONSE (FIR)'/
     1 25X, 'LINEAR PHASE DIGITAL FILTER DESIGN'/
     2 25X, 'REMEZ EXCHANGE ALGORITHM'/)
      PRINT 365
  365 FORMAT(25X, 'BANDPASS FILTER'/)
      PRINT 378, NFILT
  378 FORMAT(15X, 'FILTER LENGTH = ',I5/)
      PRINT 380
  380 FORMAT(15X,'***** IMPULSE RESPONSE *****')
      DO 381 J=1,NFCNS
      K=NFILT+1-J
      PRINT 382,J,H(J),K
381   CONTINUE
382   FORMAT(20X,'H(',I3,')=',E15.8,' = H(',I4,')')
      DO 450 K=1,NBANDS,4
      KUP=K+3
      IF(KUP.GT.NBANDS) KUP=NBANDS
      PRINT 385, (J,J=K,KUP)
  385 FORMAT(/24X,4('BAND',I3,8X))
      PRINT 390, (EDGE(2*J-1),J=K,KUP)
  390 FORMAT(2X,'LOWER BAND EDGE',5F15.9)
      PRINT 395, (EDGE(2*J),J=K,KUP)
  395 FORMAT(2X,'UPPER BAND EDGE',5F15.9)
      PRINT 400, (FX(J),J=K,KUP)
  400 FORMAT(2X,'DESIRED VALUE',2X,5F15.9)
      PRINT 410, (WTX(J),J=K,KUP)
  410 FORMAT(2X,'WEIGHTING',6X,5F15.9)
      DO 420 J=K,KUP
  420 DEVIAT(J)=DEV/WTX(J)
      PRINT 425,(DEVIAT(J),J=K,KUP)
```

```
425 FORMAT(2X,'DEVIATION',6X,5F15.9)
    DO 430 J=K,KUP
430 DEVIAT(J)=20.0*ALOG10(DEVIAT(J))
    PRINT 435, (DEVIAT(J),J=K,KUP)
435 FORMAT(2X, 'DEVIATION IN DB',5F15.9)
450 CONTINUE
    PI2=P12
    DO 452 J=1,NZ
    AMP(J)=H(NFCNS)
    FRE(J)=GRID(IEXT(J))
    DO 451 NN=1,NM1
    AMP(J)=AMP(J)+2.*H(NM1-NN+1)*COS(FRE(J)*PI2*FLOAT(NN))
451 CONTINUE
452 CONTINUE
    PRINT 455, (FRE(J),J=1,NZ)
455 FORMAT(2X,'EXTREMAL FREQUENCIES'/(2X,8F12.7))
    PRINT 456, (AMP(J),J=1,NZ)
456 FORMAT(2X,'MAGNITUDE OF FREQUENCY RESPONSES'/(2X,8F12.7))
    PRINT 460
460 FORMAT(/1X, 70(1H*)/1H1)
    NPT=2*NBANDS
    DO 470 J=1,NPT
    FXA(J)=FX((J+1)/2)
470 CONTINUE
    CALL PLTDTA(FRE,AMP,NZ,EDGE,FXA,NPT)
700 RETURN
    END
```

```
C       GEE.001            06-NOV-79
C
        DOUBLE PRECISION FUNCTION GEE(N,BLIP,X,AD,Y)
        DIMENSION X(1),Y(1),AD(1)
        DOUBLE PRECISION P12,X,Y,AD
        P12=6.283185307179586
        P=0.
        XF=BLIP
        XF=DCOS(P12*XF)
        D=0.
        DO 1 J=1,N
        O=XF-X(J)
        O=AD(J)/O
        D=D+O
1       P=P+O*Y(J)
        GEE=P/D
        RETURN
        END
```

APPENDIX C

SAMPLE OUTPUT FROM PROGRAM DESIGN

```
**********************************************************************

                 FINITE IMPULSE RESPONSE (FIR)
                 LINEAR PHASE DIGITAL FILTER DESIGN
                 REMEZ EXCHANGE ALGORITHIM

                 BANDPASS FILTER

         FILTER LENGTH= 33

         ***** IMPULSE RESPONSE *****
              H(  1)= -.43343124E-02 = H(  33)
              H(  2)=  .28107133E-01 = H(  32)
              H(  3)=  .36576607E-01 = H(  31)
              H(  4)=  .48062215E-01 = H(  30)
              H(  5)=  .52151146E-01 = H(  29)
              H(  6)=  .44314948E-01 = H(  28)
              H(  7)=  .24178276E-01 = H(  27)
              H(  8)= -.39524089E-02 = H(  26)
              H(  9)= -.31592790E-01 = H(  25)
              H( 10)= -.48276263E-01 = H(  24)
              H( 11)= -.44888714E-01 = H(  23)
              H( 12)= -.17057197E-01 = H(  22)
              H( 13)=  .32779728E-01 = H(  21)
              H( 14)=  .95199035E-01 = H(  20)
              H( 15)=  .15595394E+00 = H(  19)
              H( 16)=  .20005638E+00 = H(  18)
              H( 17)=  .21618581E+00 = H(  17)
```

```
                          BAND  1        BAND  2         BAND
LOWER BAND EDGE     0.000000000     .120000000
UPPER BAND EDGE      .100000000     .500000000
DESIRED VALUE       1.000000000    0.000000000
WEIGHTING          10.000000000  100.000000000
DEVIATION            .350741255     .035074126
DEVIATION IN DB-.910026296D+01-.291002630D+02

EXTREMAL FREQUENCIES
   0.0000000    .0367647    .0753676    .1000000    .1200000
    .1310294    .1549265    .1825000    .2137500    .2450000
    .2762500    .3075000    .3387500    .3718382    .4030882
    .4361765    .4674265    .5000000

**********************************************************************
```

DISTRIBUTION LIST

No. of Copies	Organization
12	Commander Defense Technical Info Center ATTN: DDC-DDA Cameron Station Alexandria, VA 22314
1	Director Defense Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, VA 22209
1	Director Defense Nuclear Agency Arlington, VA 22209
1	Commander US Army BMD Advanced Technology Center ATTN: BMDATC-M P. O. Box 1500 Huntsville, AL 35807
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDMD-ST 5001 Eisenhower Avenue Alexandria, VA 22333
2	Commander US Army Armament Research and Development Command ATTN: DRDAR-TSS Dover, NJ 07801
1	Commander US Army Armament Research and Development Command ATTN: L. Goldsmith Dover, NJ 07801
9	Commander US Army Armament Research and Development Command ATTN: DRDAR-LC, T. Moore W. Williver K. Reuben H. Fair S. Bernstein G. Demitrack W. Benson B. Knutulski G. Bubb Dover, NJ 07801
2	Commander US Army Armament Research and Development Command ATTN: DRDAR-SC, B. Shulman Mr. Webster Dover, NJ 07801
2	Commander US Army Armament Research and Development Command ATTN: DRDAR-SE Dover, NJ 07801
1	Commander US Army Armament Research and Development Command ATTN: DRDAR-FU Dover, NJ 07801
2	Commander US Army Armament Research and Development Command ATTN: DRDAR-DP Dover, NJ 07801
2	Commander US Army Armament Research and Development Command ATTN: DRDAR-QA Dover, NJ 07801

DISTRIBUTION LIST

No. of Copies	Organization
1	Commander US Army Armament Materiel Readiness Command ATTN: DRSAR-LEP-L, Tech Lib Rock Island, IL 61299
1	Director US Army ARRADCOM Benet Weapons Laboratory ATTN: DRDAR-LCB-TL Watervliet, NY 12189
2	Director US Army ARRADCOM Benet Weapons Laboratory ATTN: DRDAR-LCB, T. Simkins T. Davidson Watervliet, NY 12189
1	Commander US Army Aviation Research and Development Command ATTN: DRDAV-E 4300 Goodfellow Blvd. St. Louis, MO 63120
2	Director US Army Research and Technology Laboratories (AVRADCOM) Ames Research Center Moffett Field, CA 94035
1	Director US Army Air Mobility Research and Development Laboratory Ames Research Center Moffett Field, CA 94035
1	Commander US Army Communications Research and Development Command ATTN: DRDCO-PPA-SA Fort Monmouth, NJ 07703
1	Commander US Army Electronics Research and Development Command Technical Support Activity ATTN: DELSD-L Fort Monmouth, NJ 07703
1	Commander US Army Harry Diamond Labs 2800 Powder Mill Road Adelphi, MD 20783
1	Commander US Army Missile Command ATTN: DRSMI-R Redstone Arsenal, AL 35809
1	Commander US Army Missile Command ATTN: DRSMI-RBL Redstone Arsenal, AL 35809
1	Commander US Army Missile Command ATTN: DRSMI-YDL Redstone Arsenal, AL 35809
2	Commander US Army Mobility Equipment Research & Development Command Fort Belvoir, VA 22060
2	Commander US Army Tank Automotive Research and Development Command ATTN: DRDTA-UL Technical Director Warren, MI 48090
2	Project Manager Division Air Defense Gun ATTN: DRCPM-ADG Dover, NJ 07801

DISTRIBUTION LIST

No. of Copies	Organization
2	Project Manager Cannon Artillery Weapons System ATTN: DRCPM-CAWS Dover, NJ 07801
2	Project Manager Nuclear Munitions ATTN: DRCPM-NUC Dover, NJ 07801
2	Project Manager Tank Main Armament Systems ATTN: DRCPM-TMA Dover, NJ 07801
1	Product Manager for 30mm Ammo. ATTN: DRCPM-AAH-30mm Dover, NJ 07801
2	Product Manager M110E2 Weapon System, DARCM ATTN: DRCPM-M110E2 Rock Island, IL 61299
3	Commander US Army Research Office P.O. Box 12211 ATTN: Technical Director Engineering Division Metallurgy & Materials Division Research Triangle Park NC 27709
1	Commander US Army Research Office ATTN: Dr. J. Chandra Research Triangle Park NC 27709
4	Director US Army Mechanics and Materials Research Center ATTN: Director (3 cys) DRXMR-ATL (1 cy) Watertown, MA 02172

No. of Copies	Organization
1	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-SL, Tech Lib White Sands Missile Range NM 88002
1	Commander Naval Air Systems Command Washington, DC 20360
2	Commander Naval Ordnance Systems Command Washington, DC 20360
1	Commander Naval Sea Systems Command Washington, DC 20362
2	Commander Naval Ship Research and Development Command Bethesda, MD 20084
1	Commander Naval Air Developmentment Center, Johnsville Warminster, PA 18974
1	Commander Naval Missile Center Point Mugu, CA 95041
2	Commander Naval Surface Weapons Center Dahlgren, VA 22448
2	Commander Naval Surface Weapons Center Silver Spring, MD 20910
2	Commander Naval Weapons Center China Lake, CA 93555

DISTRIBUTION LIST

No. of Copies	Organization
1	Commander Naval Research Laboratory Washington, DC 20375
1	Superintendent Naval Postgraduate School ATTN: Dir of Lib Monterey, CA 93940
1	Commander Naval Ordnance Station Indian Head, MD 20640
2	AFRPL ATTN: W. Andrepont T. Park Edwards AFB, CA 93523
2	AFATL Eglin AFB, FL 32542
2	AFWL Kirkland AFB NM 87117
2	ASD Wright-Patterson AFB OH 45433
1	Director Lawrence Livermore Laboratory Livermore, CA 94550
1	Director Los Alamos Scientific Laboratory Los Alamos, NM 87544
1	Headquarters National Aeronautics and Space Administration Washington, DC 20546
2	Director National Aeronautics and Space Administration Langley Research Center Hampton, VA 23365
1	Director National Aeronautics and Space Administration Manned Spacecraft Center ATTN: Lib Houston, TX 77058
1	BLM Applied Mechanics Consultants ATTN: Dr. A. Boresi 3310 Willett Drive Laramie, WY 82070
1	CALSPAN Corp. ATTN: E. Fisher P. O. Box 400 Buffalo, NY 14225
1	S&D Dynamics, Inc. ATTN: Dr. M. Soifer 755 New York Avenue Huntington, NY 11743
1	Southwest Research Institute ATTN: P. Cox 8500 Culebra Road San Antonio, TX 78228
1	Stanford University Stanford Linear Accelerator Center ATTN: Eric Grosse, Numerical Analysis Consultant ALAC, P. O. Box 4349 Stanford, CA 94305
1	Towson State University Department of Mathematics ATTN: Miss Margaret Zipp Towson, MD 21204

DISTRIBUTION LIST

No. of Copies	Organization
2	University of Delaware Department of Mathematics Department of Mechanical Engr. Newark, DE 19711
1	University of Illinois Department of Mathematics ATTN: Dr. Evelyn Frank Urbana, IL 61801
1	University of Kentucky Department of Computer Science ATTN: Prof. H. C. Thacher, Jr. 915 Patterson Office Tower Lexington, KY 40506
3	University of Wisconsin-Madison Mathematics Research Center ATTN: Dr. John Nohel Dr. Carl de Boor Dr. George Box 610 Walnut Street Madison, WI 53706
1	Virginia Commonwealth Univ. Department of Math. Sciences ATTN: Mr. V. Benokraitis 901 W. Franklin Richmond, VA 23284
1	Virginia Polytechnic Institute Dept. of Aerospace Engineering ATTN: Prof. G. Inger Blacksburg, VA 24061

No. of Copies	Organization
	Aberdeen Proving Ground
2	Dir, USAMSAA ATTN: DRXSY-D DRXSY-MP, H. Cohen
2	Dir, MTD ATTN: H. King P. Paules
1	Cdr, USATECOM ATTN: DRSTE-TO-F
6	Cdr/Dir USACSL, EA ATTN: DRDAR-CL DRDAR-CLB DRDAR-CLB-PA DRDAR-CLD DRDAR-CLN DRDAR-CLY

USER EVALUATION OF REPORT

Please take a few minutes to answer the questions below; tear out this sheet, fold as indicated, staple or tape closed, and place in the mail. Your comments will provide us with information for improving future reports.

1. BRL Report Number ______________________________

2. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)

____

____

____

3. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.) ______________________________

____

____

4. Has the information in this report led to any quantitative savings as far as man-hours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate.

____

____

5. General Comments (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.) ______________

____

____

____

6. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

Name: ______________________________

Telephone Number: ______________________________

Organization Address: ______________________________

____

____